

RESEARCH ARTICLE

Open Access

Analysing RNA-kinetics based on folding space abstraction

Jiabin Huang and Björn Voß*

Abstract

Background: RNA molecules, especially non-coding RNAs, play vital roles in the cell and their biological functions are mostly determined by structural properties. Often, these properties are related to dynamic changes in the structure, as in the case of riboswitches, and thus the analysis of RNA folding kinetics is crucial for their study. Exact approaches to kinetic folding are computationally expensive and, thus, limited to short sequences. In a previous study, we introduced a position-specific abstraction based on helices which we termed helix index shapes (*hishapes*) and a *hishape*-based algorithm for near-optimal folding pathway computation, called HiPATH. The combination of these approaches provides an abstract view of the folding space that offers information about the global features.

Results: In this paper we present HiKINETICS, an algorithm that can predict RNA folding kinetics for sequences up to several hundred nucleotides long. This algorithm is based on RNAHELICES, which decomposes the folding space into abstract classes, namely *hishapes*, and an improved version of HiPATH, namely HiPATH2, which estimates plausible folding pathways that connect these classes. Furthermore, we analyse the relationship of *hishapes* to locally optimal structures, the results of which strengthen the use of the *hishape* abstraction for studying folding kinetics. Finally, we show the application of HiKINETICS to the folding kinetics of two well-studied RNAs.

Conclusions: HiKINETICS can calculate kinetic folding based on a novel *hishape* decomposition. HiKINETICS, together with HiPATH2 and RNAHELICES, is available for download at <http://www.cyanolab.de/software/RNAHeliCes.htm>.

Keywords: RNA, Folding space, Kinetics, Abstraction

Background

RNA molecules play vital roles in the cell, and their function is often determined by structural properties. These properties may be static, such as structural motifs, or dynamic, such as the ability to adopt different conformations as riboswitches do. The latter emphasises the importance of studying RNA folding kinetics, which is the dynamic behaviour of RNA structure over time.

Most approaches to the stochastic simulation of RNA folding kinetics can be described as Monte Carlo simulations [1-3] or continuous time Markov chains (CTMC) [4,5]. A Monte Carlo simulation requires a large number of samples of individual trajectories to achieve accuracy, rendering these methods computationally expensive. The same holds true for CTMC-based simulation, as long as it is based on a complete enumeration of the folding space.

The program TREEKIN [4] implements a CTMC-based simulation, and for short sequences (e.g., up to 30 nt), can simulate exact folding kinetics. For longer sequences, however, the exponential growth of the underlying state space requires restricting the analysis to a subset of the folding space. For this purpose so called “macrostates” were introduced in [4], each of which can be seen as a local minimum and all structures that are connected to it by a gradient walk. A macrostate is represented by its local minimum secondary structure. The problem that arises from the macrostate definition is that neighbouring macrostates cannot easily be identified. The program TREEKIN uses BARRIERS to compute saddle points connecting macrostates and the corresponding transition rates. The dependence on BARRIERS limits this approach to sequences of moderate length (up to 100 nt), which can be partially overcome by restricting the analysis to conformations within a specified energy range above the minimum free energy. To overcome the length restriction and reduce the computational burden Tang et al. [6] use a

*Correspondence: bjoern.voss@biologie.uni-freiburg.de
Genetics & Experimental Bioinformatics, Faculty of Biology, University of Freiburg, Schänzlestr. 1, 79104, Freiburg, Germany

sampling strategy called probabilistic Boltzmann-filtered suboptimal sampling. In their approach, sampled structures are connected by transition paths computed using a simple greedy algorithm [7]. These transition paths are weighted with their barrier energy. The procedure may be suboptimal in two ways: first, the sampling may miss important structures in the folding space, and second, the greedy pathway prediction may overestimate energy barriers and lead to inaccurate transition rates.

The computation of exact, globally optimal folding pathways between any two secondary structures (e.g., BARRIERS [1,8]) is NP-hard [9]. Many heuristic approaches for computing folding pathways have been proposed. The first approach was proposed by Morgan and Higgs [10] by selecting the least “clashing” base-pairs as the next intermediate structure from a set of neighbouring structures. Subsequently, the idea was extended by Flamm *et al.* [11]. Instead of selecting the best structure as the next intermediate structure, the k best candidates are maintained during the folding pathway construction (breadth first search, BFS). In contrast to these direct path heuristics (intermediate structures contain only base pairs that are also present in the start or target structure), Dotu *et al.* [12] presented a heuristic including indirect paths. Li *et al.* [13] proposed an evolutionary algorithm in which a pathway is represented by an action chain that is mutated by different strategies to find a better solution.

In general there are two central challenges in CTMC-based folding simulations for RNA. How can the energy landscape be decomposed in a complete, compact and non-heuristic way? And how can the transition rates between partitions be calculated accurately and efficiently?

Our contributions in this paper address these challenges. In previous work [14], we introduced *hishapes* as classes of structures sharing the same helices. These *hishapes* intrinsically decompose the folding space into disjoint classes, which are represented by the member with minimum free energy, called the *hishrep*. This partitioning is complete and non-heuristic, and its coarse-graining can be adjusted based on its abstraction levels, which differ in the type of structural elements they consider. Here, we analyse the degree to which *hishapes* overlap with locally optimal structures. Additionally, we provide a new folding space restriction, called strictly negative structures, that eliminates suboptimal structures with positive energy substructures. We present HIPATH2 as an improved version of HIPATH [14] and show that it computes lower energy barrier folding pathways for most cases in our benchmark set. Finally, we combine these methods in HIKINETICS, a tool for simulating RNA folding kinetics using strictly negative *hishapes* for the folding space decomposition and energy barriers estimated by HIPATH2 to derive transition rates using Arrhenius’

equation. We apply our novel kinetic analysis tool termed HIKINETICS to two well-studied RNAs.

Results and discussion

Hishapes revisited

We begin with a brief recapitulation of the central concepts and notations of *hishapes*. For formal definitions, we refer the reader to our previous manuscript [14]. For *hishapes*, we consider an RNA secondary structure as a set of helices terminated by loops (internal, bulge, multiple and hairpin loops). The innermost base pair (i, j) of a helix corresponds to the closing base pair of the terminating loop, and we define $(j - i)/2$ to be the *helix index* of this helix. Additionally, we mark the *helix index* with m , b , or i for multiple, bulge, or internal loop, respectively. Using a mapping function π , we can now map any secondary structure to a *helix index* shape (*hishape*), which is simply a list of *helix indices*. Figure 1 illustrates the relationship among helices, *helix indices* and *hishapes*. To provide different levels of abstraction, we make use of different mapping functions. The function π_h retains only hairpin loop helices and π_{h+} additionally keeps track of the nesting within multiloops. The functions π_m and π_a extend π_{h+} through retaining multiloops and all helices, respectively. A *hishape* defines a class of similar structures, and we use the member with minimum free energy as the *hishape* representative (*hishrep*).

Reducing the search space to strictly negative structures

The number of feasible secondary structures grows exponentially with the length of the RNA. We recently presented *hishapes*, which abstract from helix lengths and, depending on the abstraction type, also from certain loop types. Compared to suboptimal structures, the number of possible *hishapes* is dramatically reduced, but it still grows exponentially with sequence length.

Hishapes provide deep insight into the folding space of an RNA molecule while keeping the output at a manageable size. Analysing one of our favourite toy examples, the Spliced Leader RNA from *Leptomonas collosoma*, we recognised that there are pairs of *hishapes* where the *hishrep* with an additional helix has a higher energy, as shown in Figure 2. Here, due to the additional helix with *helix index* 13, the energy of *hishape* [13, 38] is worse than the energy of *hishape* [38].

The formation of this helix imposes an energy cost of 1.2 kcal/mol and, thus, is thermodynamically unfavourable. To eliminate such unfavourable structures, we cannot simply exclude all positive energy substructures within our recursive DP calculation. Doing so would for example disallow nearly all hairpin loops and thereby the computation of many biologically significant structures. We take the view that closed substructures within the external loop or within a multiloop must not have positive energy.

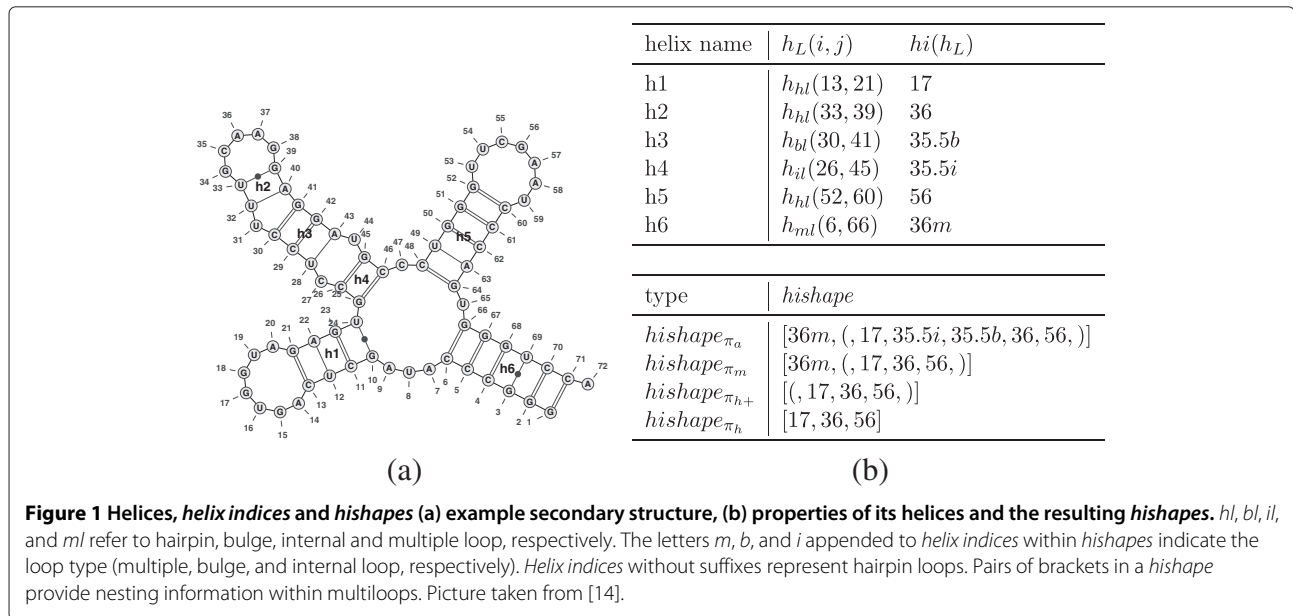


Figure 1 Helices, helix indices and hishapes (a) example secondary structure, (b) properties of its helices and the resulting hishapes. *hl*, *bl*, *il*, and *ml* refer to hairpin, bulge, internal and multiple loop, respectively. The letters *m*, *b*, and *i* appended to *helix indices* within *hishapes* indicate the loop type (multiple, bulge, and internal loop, respectively). *Helix indices* without suffixes represent hairpin loops. Pairs of brackets in a *hishape* provide nesting information within multiloops. Picture taken from [14].

We are aware that disallowing positive energy substructures within multiloops may even remove the minimum free energy (MFE) structure from the structure space. In fact, a test on 10,000 randomly selected sequences from Rfam showed that for 1.67% of the sequences, the MFE structure is removed. For these 167 sequences, the strictly negative optimal structure has a ΔG that is on average 0.49 kcal/mol ($\sigma = 0.367$, $max = 2.3$ kcal/mol) worse than the MFE. However, these differences are on the same scale as (or even below) the uncertainties present in the thermodynamic parameters used for computation.

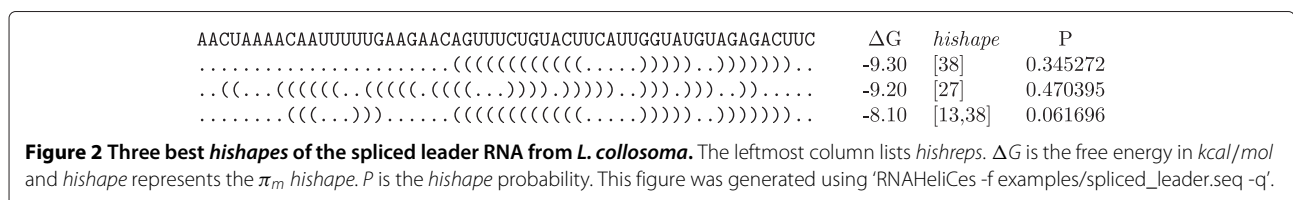
A further reason we think that removing substructures with positive energy is reasonable is that they seem kinetically unfavourable. A helix nucleates by formation of the terminal hairpin loop, which is the time dominating step, and is subsequently stabilised by the stacking of base pairs. For positive energy substructures, the ΔG of the hairpin loop is very large, which results in a low probability of nucleation, and/or the ΔG of the stacking pairs is small, which renders the melting of such helices very likely. For these reasons, we believe that disallowing positive energy substructures is a reasonable method to reduce the search space, although it is a heuristic filtering.

Because we can check for substructures with positive energy during the recursive calculation, this filter has

nearly no computational burden. On the contrary, the reduced number of intermediate results actually speeds up the calculation. Restricting the analysis to strictly negative (SN) *hishapes* significantly reduces the search space (see Figure 3). It still grows exponentially with sequence length, but much more slowly, which is reflected by the much smaller base in the exponential growth asymptotics.

Hishreps versus local optimal structures

We were interested in the question of to what extent *hishreps* overlap with the set of locally optimal structures. As described, e.g., in [16], a locally optimal structure has the lowest free energy compared with its neighbouring structures, which are the structures that differ from it by a single base pair. Because our approach disregards any structure that contains isolated base pairs, we slightly modify the concept of the neighbourhood. Commonly, a neighbour (A') of the observed structure (A) is defined by adding (or deleting) a base pair in A . This definition also holds true for our purposes, as long as A' does not carry a lonely base pair. If A' does contain a single lonely base pair as the result of previously removing a base pair, then we also delete the isolated one, resulting in the structure (A''), which will still be treated as a neighbour of A . Vice versa, if A' carries an isolated base pair due to its addition we close,



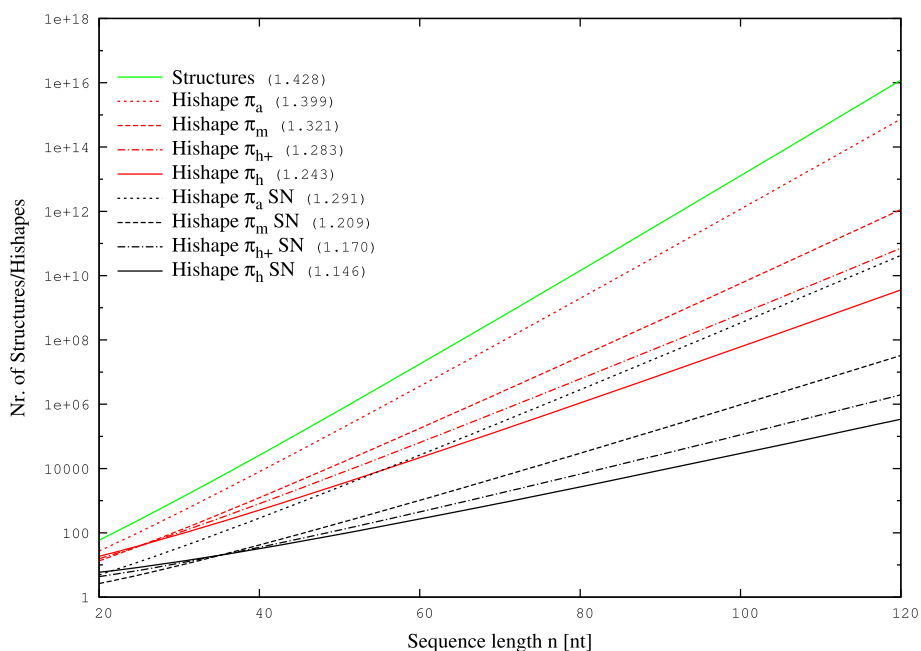


Figure 3 Comparison of structure/hishape spaces. All possible structures and *hishapes* were predicted for random sequences of lengths ranging from 20-120 nt, using RNASUBOPT-noLP and RNAHELICES with different abstraction levels and restricting to strictly negative (SN) structures, respectively. The average numbers of structures/*hishapes* for each length were fitted to the formula $a \times b^n \times n^{-3/2}$ [15]. The numbers in parentheses give the values for b , which is the dominating factor in this term.

if possible, an adjacent base pair. The resulting structure A'' is then a neighbour to A . Note that in the two described cases, A and A'' differ by two adjacent base pairs. This version of the neighbourhood should be essentially the same as the ‘noLP’ move set from BARRIERS.

Based on this definition, we check whether our predicted *hishreps* are locally optimal or not. Table 1 shows, for the different abstraction levels and for strictly negative *hishapes* and all *hishapes*, the fractions of *hishreps* that are local optima. Overall, the fractions are quite high, sometimes reaching 100%. The sequence for the S-box leader constitutes a negative outlier, especially in the case of strictly negative structures, where at most only 15% of the π_h *hishreps* are locally optimal. Strikingly, strictly negative *hishreps* less frequently correspond to local minima compared to the unrestricted case. This result is somewhat counterintuitive but may be explained as follows. Filtering for strictly negative *hishapes* removes many *hishapes*. Because most *hishapes* are actually local minima, as can be seen for the unfiltered version, these *hishapes* are also affected the most strongly. Thus, the fraction of non-local optima increases in the case of strictly negative *hishapes*. So what are these non-locally optimal *hishreps*? In our opinion, they are mainly the result of replacing helices by single stranded regions. Because the formation of the removed helix would result in a neighbouring structure with better energy, the *hishrep* of the resulting *hishape* is not a local minimum.

This reasoning together with the fact that in abstraction type π_a the largest number of helices is taken into account, also explains to a large degree why *hishreps* for abstraction type π_a are less often locally optimal than *hishreps* of types π_m , π_{h+} and π_h .

The opposite question, “do all locally optimal structures belong to distinct *hishapes*” is easier to address. For abstractions π_m , π_{h+} and π_h the structures do not have to belong to distinct *hishapes* as two locally optimal structures differing, e.g., by an internal loop, will be mapped to the same *hishape*. The situation is different for π_a *hishapes*, as they account for differences in all loop types. Starting from any locally optimal structure, the extension and shortening of helices cannot lead to another locally optimal structure. Reaching another locally optimal structure is only possible by adding or removing complete helices or by helix interruption, i.e., the introduction of internal or bulge loops. All these events will introduce new helices into the π_a abstraction, thus resulting in different *hishapes*. This point is nicely reflected by the fractions of locally optimal structures that are also *hishreps* ($\frac{|H \cap C|}{|C|}$, Table 1). While locally optimal structures have a fairly high overlap with *hishreps* of the least abstract types π_a and π_a^{SN} , the overlap drops significantly for the other abstraction types, as many local optima differ in the composition of their internal and bulge loops and are thus not retained on these abstraction levels, as described above.

Table 1 Fractions of locally optimal *hishreps*

Instance	Length	π_a	π_m	π_{h+}	π_h	π_a^{SN}	π_m^{SN}	π_{h+}^{SN}	π_h^{SN}
					[%]				
SL	56 nt	79.00	98.00	98.00	99.00	73.00	85.00	94.29	96.88
		90.80	24.32	21.17	18.37	56.59	2.63	2.56	2.40
attenuator	73 nt	81.00	100.00	100.00	100.00	77.00	99.00	96.00	92.00
		94.19	32.15	23.98	18.90	76.24	14.41	2.38	0.95
ms2	73 nt	84.00	98.00	98.00	97.00	82.00	89.00	80.00	81.00
		91.30	15.96	11.84	10.85	79.61	1.31	0.38	0.29
s15	74 nt	87.00	100.00	100.00	100.00	82.00	96.00	97.00	100.00
		90.63	16.45	13.30	10.96	73.21	4.23	1.28	0.85
dsrA	85 nt	77.00	98.00	98.00	99.00	71.00	97.00	98.00	100.00
		83.70	27.30	22.58	16.39	57.26	4.56	2.00	0.76
rb2	113 nt	76.00	92.00	92.00	93.00	75.00	88.00	88.00	85.00
		79.17	28.13	27.38	22.79	74.26	12.92	11.50	7.10
alpha operon	130 nt	99.00	98.00	96.00	96.00	98.00	100.00	99.00	100.00
		96.12	9.82	4.15	2.22	72.59	1.26	0.45	0.28
rb3	141 nt	76.00	99.00	99.00	99.00	76.00	99.00	99.00	98.00
		96.20	30.56	21.48	17.52	96.20	23.24	10.61	8.11
amv	145 nt	77.00	89.00	89.00	83.00	78.00	89.00	89.00	81.00
		82.80	38.03	38.03	4.56	83.87	38.03	38.03	4.45
rb4	146 nt	96.00	100.00	100.00	100.00	97.00	100.00	99.00	81.00
		89.72	10.80	8.04	5.03	20.51	2.35	1.36	0.88
rb1	148 nt	86.00	100.00	100.00	100.00	81.00	100.00	99.00	98.00
		72.27	8.87	6.61	5.67	61.36	4.84	1.66	1.21
HDV	153 nt	96.00	100.00	100.00	100.00	96.00	100.00	99.00	98.00
		87.27	37.04	5.62	2.64	87.27	26.74	2.61	1.00
thiM leader	165 nt	85.00	100.00	100.00	100.00	82.00	100.00	100.00	100.00
		98.84	24.33	15.72	9.00	86.32	18.38	7.94	5.01
rb5	201 nt	100.00	100.00	99.00	99.00	100.00	100.00	99.00	97.00
		93.46	29.07	4.06	2.24	93.46	25.06	2.24	0.78
sbox leader	247 nt	77.00	76.00	68.00	60.00	68.00	65.00	42.00	15.00
		91.67	28.36	15.70	6.76	58.62	14.84	3.11	0.42
HIV-1 leader	280 nt	39.00	71.00	58.00	56.00	38.00	52.00	38.00	38.00
		48.75	2.09	1.34	1.21	47.50	1.20	0.60	0.54
ribD leader	304 nt	91.00	100.00	85.00	85.00	88.00	92.00	76.00	70.00
		81.25	37.17	8.83	5.53	78.57	29.21	6.23	3.18
hok	396 nt	58.00	79.00	69.00	62.00	57.00	83.00	72.00	65.00
		53.70	1.42	0.70	0.51	52.78	1.38	0.60	0.42

In each cell, the upper number represents the fraction of the set of *hishreps* \mathcal{H} that are also locally optimal $\left(\frac{|\mathcal{H} \cap \mathcal{L}|}{|\mathcal{H}|}\right)$ and the lower number represents the fraction of the set of local optima that are also *hishreps* $\left(\frac{|\mathcal{H} \cap \mathcal{L}|}{|\mathcal{L}|}\right)$. We restricted the computation of *hishreps* to the best 100 and the computation of the local optima to the corresponding energy range $\max\{\Delta G(x) : x \in \mathcal{H}\}$ above the MFE. The dataset is taken from [12].^{SN} strictly negative *hishapes*.

Improved barrier energy estimation

Pathways connecting alternative structures are important features of the folding space, especially when studying

folding kinetics. Here, transition rates computed based on the energy barriers, which are derived from the pathways between structures, are commonly used. It has been

shown that the problem of computing the globally optimal folding pathway between two structures is NP-hard [9]. In our recent publication [14], we provided an overview of current pathway estimation tools and introduced HiPATH, outperforming the other analysed methods. Here, we present an improved version, which we term HiPATH2. One of the essential features of HiPATH is that it uses a set of related *hishapes* as anchors for estimating a (near-) optimal pathway between two structures. These related *hishapes* correspond to *hishapes* that consist of individual helix indices from the start and target structures or combinations thereof. By detailed inspection of the optimal folding pathways obtained by BARRIERS, we observed that pathway intermediates sometimes carry helices with helix indices that are not identical, but very similar to the helix indices of the start or target *hishape*, differing by only a few positions. Therefore, we implemented fuzzy related *hishapes* that also take into account the neighbourhoods (in terms of the *helix index* distance) of related *hishapes*.

HiPATH2, which is based on fuzzy related *hishapes* was benchmarked against existing methods (BARRIERS [1,8], BFS [11], RNATABUPATH [12], RNAEAPATH [13] and HiPATH [14]) on 18 conformational switches taken

from [12] (see Table 2). They consist of two parts: five of them are riboswitches (rb1, rb2, rb3, rb4 and rb5) taken from [17,18], and the remaining 13 are taken from PARNASS [19]. All of the algorithms were used with the same energy rules (Turner99) [20,21]. We use the “microstate” grammar [22], which corresponds to the “-d1” option of RNAEVAL from the Vienna RNA package [23]. All other parameters were kept as the defaults.

The results in Table 2 show that in most cases, HiPATH2, together with other methods, produces the lowest energy barrier estimates. In the four cases where exact pathways are known, the sum of errors is reduced from 1.7 to 0.8 compared to HiPATH. Compared to the second best method, RNAEAPATH, HiPATH2 produces slightly (0.1 to 0.4 kcal/mol) less optimal pathways in four cases (rb2, hok, thiM leader, HIV-1 leader). However, in eight cases it performs better by 0.14 to 2.26 kcal/mol. A major difference is found in the runtimes of the two. Table 3 compares the runtimes of HiPATH2 and RNAEAPATH. While RNAEAPATH spends approximately 837 min., HiPATH2 only needs approximately 192 min., thus being 4.4 times faster.

Table 2 Comparison of BARRIERS (BAR), BFS, RNATABUPATH (TABU), RNAEAPATH (EA) and HiPATH

Instance	Length	BAR	BFS (k = 10)	TABU (n = 1000)	EA	HiPATH	HiPATH2
rb1	148 nt	-	24.04	24.04	23.2	20.94	20.94
rb2	113 nt	*	8.2	7.25	6.5	6.6	6.6
rb3	141 nt	-	22.4	17.9	17.5	16.7	16.7
rb4	146 nt	-	16.9	16.9	16.9	16.9	16.9
rb5	201 nt	-	24.54	24.54	21.44	21.44	21.44
hok	396 nt	-	28.5	29.66	20.7	21.1	21.1
SL	56 nt	11.8	13	12.9	13	12.4	12.4
attenuator	73 nt	8.3	8.7	8.6	8.7	8.6	8.5
s15	74 nt	6.6	7.1	6.6	7.1	7.1	6.6
sbox leader	247 nt	*	5.2	5.2	5.2	5.2	5.2
thiM leader	165 nt	-	16.13	14.84	12.3	14	12.4
ms2	73 nt	*	6.6	6.6	6.6	6.6	6.6
HDV	153 nt	-	17.4	17	16.8	15.6	15.6
dsrA	85 nt	8	8.3	8.2	8	8.3	8
ribD leader	304 nt	-	10.71	9.5	9.5	10.71	9.5
amv	145 nt	*	5.8	5.8	5.74	5.8	5.5
alpha operon	130 nt	*	6.5	6.5	6.1	6.5	5.96
HIV-1 leader	280 nt	-	9.3	11.3	8.9	9.3	9.3

The dataset was taken from [12,19], the results for BFS and RNATABUPATH from [12] and the results for EA from [13]. Energy barriers are given in kcal/mol. The *maxkeep* value *k* was 10 for BFS itself and for the BFS used within HiPATH and HiPATH2. HiPATH2 was used with auto-adjusted fuzzy related *hishape* numbers, π_0 and $\theta = 1.5$. HiPATH was used with the default parameters. Bold numbers represent the minimum value for the respective sequence. The symbol "*" means BARRIERS could not be applied because either the start or the target structure was not locally optimal. The symbol "-" means computation did not finish within one day. The energy range used with RNASUBOPT for BARRIERS was determined using HiPATH2 and set to the barrier energy of HiPATH2 + 1 kcal/mol. Note that the results may be different from the ones shown in [14] since the used start and target structures may differ. Here we used the ones provided in [12], while in [14] we derived them for ourselves.

Table 3 Runtime comparison of RNAEAPATH and HiPATH2

Instance	RNAEAPATH	HiPATH2
rb1	38m 49s	8m 59s
rb2	10m 31s	5m 20s
rb3	25m 45s	11m 22s
rb4	0m 02s	5m 33s
rb5	14m 31s	14m 20s
hok	443m 51s	45m 13s
SL	12m 49s	1m 31s
attenuator	15m 56s	1m 15s
s15	11m 42s	1m 06s
sbox leader	24m 47s	19m 21s
thiM leader	48m 28s	7m 41s
ms2	15m 03s	0m 25s
HDV	30m 57s	9m 50s
dsrA	14m 51s	2m 10s
ribD leader	59m 33s	24m 20s
amv	15m 05s	10m 35s
alpha operon	16m 50s	4m 59s
HIV-1 leader	37m 54s	18m 01s
Total	837m 33s	192m 09s

Run times were measured as described before [14], and both programs were used with default parameters. Sequences were taken from [12,19], and all tests were run on an 8x AMD Opteron 8378 machine with 128 GB RAM under openSUSE 11.2 (x86_64).

Simulating folding kinetics

Our approach for simulating folding kinetics is based on a set of *hishapes* connected by pathways with their corresponding barrier energies. The most straightforward approximation of transition rates can be done using Arrhenius' equation. Consider the two *hishapes* α and β . We initially compute the *hishape* ensemble energy ($\Delta G(\alpha)$, $\Delta G(\beta)$) via the *hishapes* partition function contribution calculated by RNAHELICES (see Equation 4). Next, using HiPATH2, we estimate the barrier energy $\Delta G[\alpha, \beta]$ between the two *hishapes* of α and β . Finally, we derive the transition rates using Arrhenius' equation (see equation 5). Using the *hishape* ensemble energy can be seen as weighting the energy by the size of the *hishape* class, which takes into account that the more members a *hishape* has, the higher the probability of a transition into the *hishape*. In contrast, transition out of a large (in terms of members) *hishape* is less likely. Our approach is conceptually similar to the macrostate model introduced with TREEKIN. Here, the folding space is partitioned into macrostates, based on local minima and their basins of attraction. These macrostates are computed by the program BARRIERS, which also computes the transition rates based on the barrier energies. The latter are computed

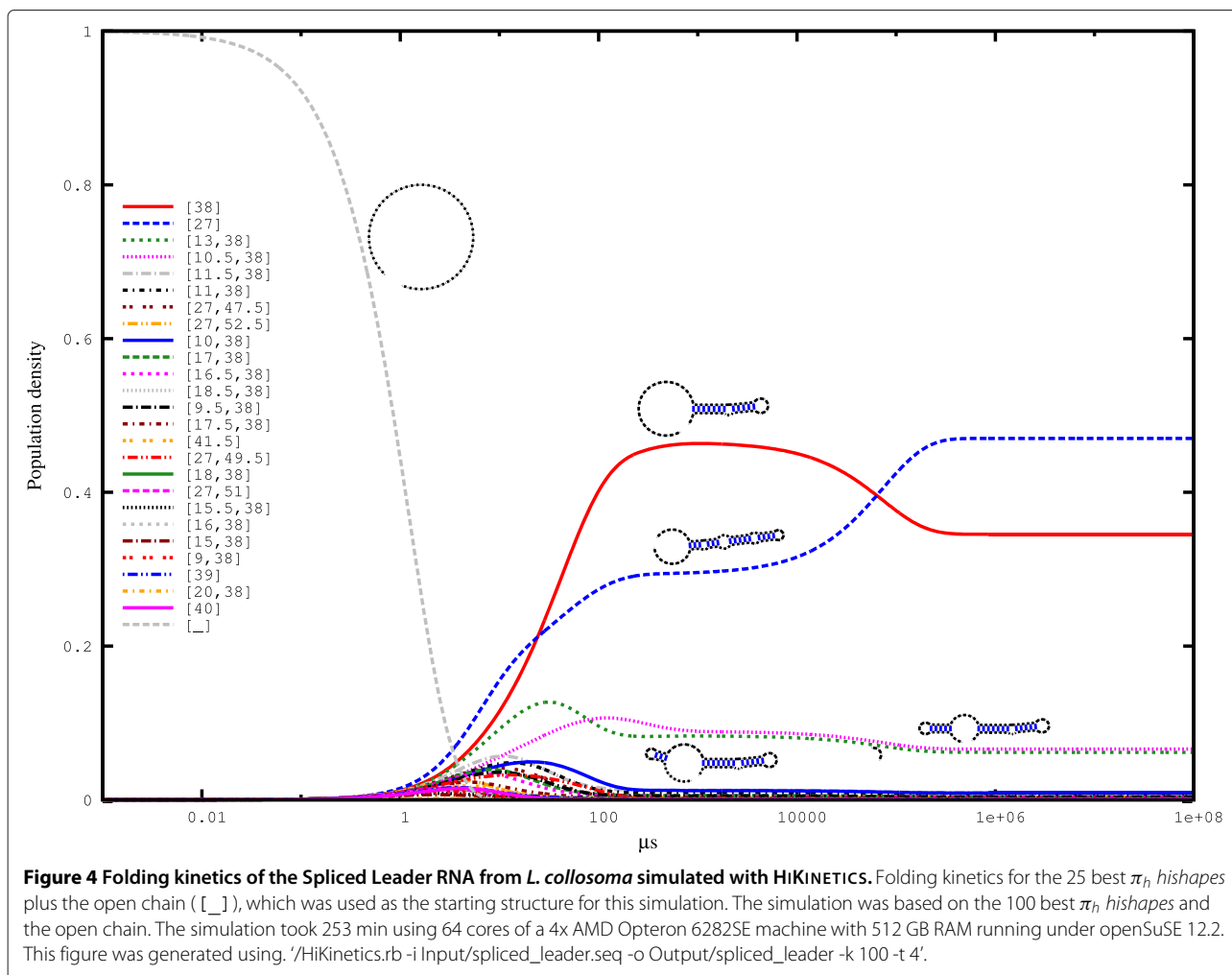
on-the-fly, which is elegant, but has one major drawback: the depth (in terms of free energy above the MFE) of the analysis must be sufficiently large to ensure that saddle points connecting all local minima (macrostates) of interest are present. For real-world examples, this depth can easily reach 10-20 kcal/mol (see Table 2), resulting in a large computational effort to compute the transition rates, especially for long sequences. Our approach circumvents this problem, as the computation of the transition rates is separated from the computation of the macrostates, i.e. *hishapes*, and the latter is more efficient, especially when restricted to strictly negative *hishapes*. Therefore, HiKINETICS is able to simulate folding kinetics for longer sequences than is possible with BARRIERS and TREEKIN. Of course, this ability does not come for free, and we expect our transition rate estimate to be less accurate than the one made using BARRIERS. The results we present in the next section show that this inaccuracy seems to have only a minor influence.

Spliced Leader RNA from *Leptomonas collosoma*

The Spliced Leader RNA from *Leptomonas collosoma* [24] has two alternating conformations of nearly equal free energy. Figure 2 shows the results of *hishape* analysis. The two π_m *hishapes* ([38] and [27]) represent the two native conformations of the Spliced Leader RNA. The probabilities of conformations 1 and 2 are 0.345271 and 0.470394, respectively, which is in agreement with the bistable character of this RNA.

The kinetic analysis in Figure 4 shows that the two major *hishapes* ([38] and [27]) dominate from $t = 10 \mu s$ until equilibrium. At the end of the simulation, their equilibrium occupancies are the same as the probability calculated by the partition function. Interestingly, both alternative *hishape* classes build plateaus that persist for a long period (from approximately $t = 500 \mu s$ to $t = 50,000 \mu s$) and cross at approximately $t = 50,000 \mu s$. If the Spliced Leader RNA degrades within this period, *hishape* [38] would be kinetically preferred, achieving almost 50% occupancy. However, if the lifetime of the Spliced Leader RNA exceeds the time needed to reach equilibrium, *hishape* [27] would win.

To determine the degree to which strictly negative filtering influences the analysis, we performed a simulation based on strictly negative *hishapes* on the same sequence (see Figure 5). Here, the (arbitrary) timescale of the process is altered, while the characteristics are the same. Note that the two *hishapes* ([13,38] and [10.5,38]), which are related to [38], are not strictly negative and thus are no longer present. As a result of the filtering, the equilibrium probabilities are also altered from 0.345 to 0.422 for *hishape* [38] and from 0.470 to 0.575 for *hishape* [27]. This result is mainly due to the reduced state space, such that each state occurs with higher frequency. Direct



computation of the probabilities for the strictly negative hishapes using RNAHELICES results in the same values.

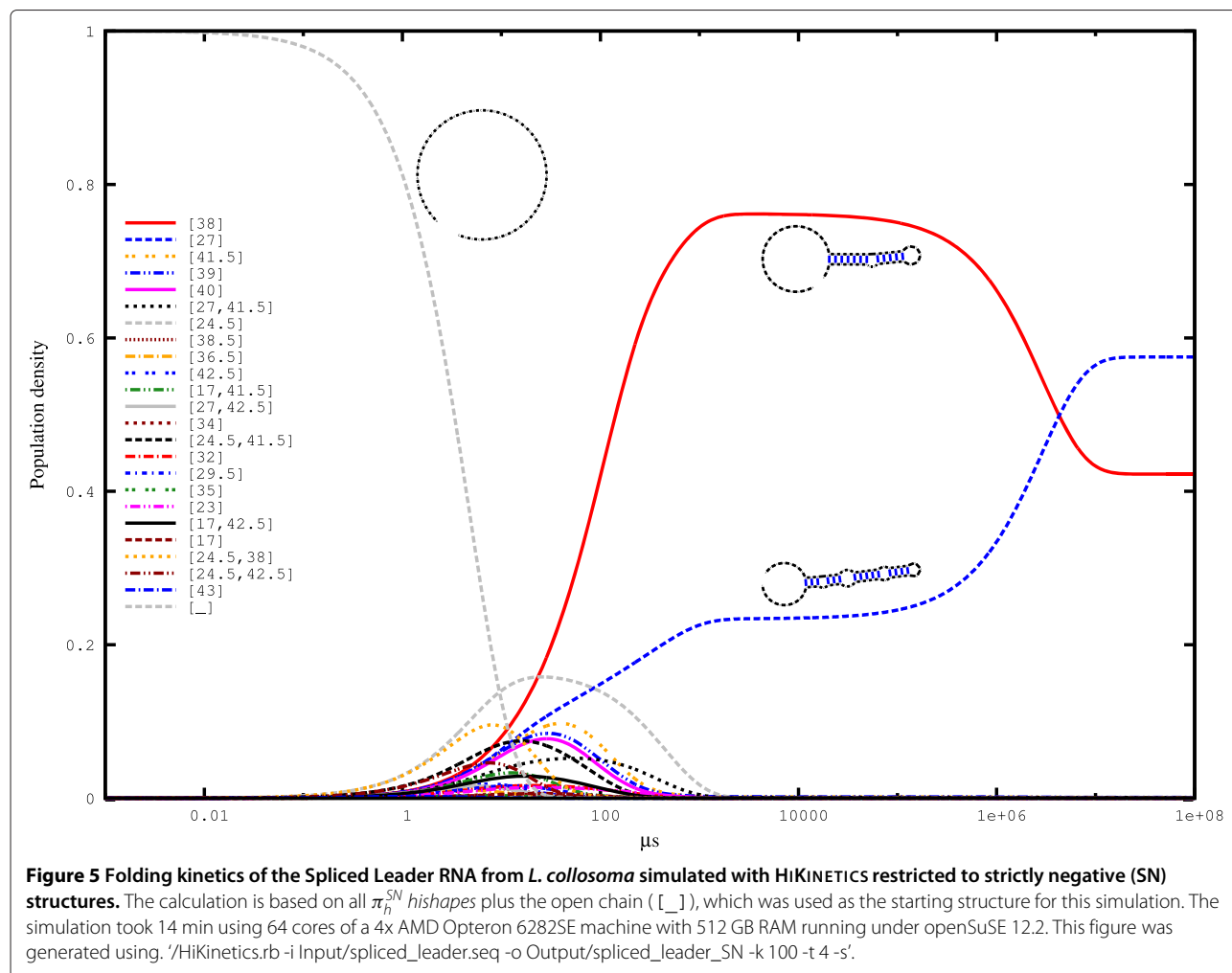
Next, we compared our hishape-based kinetics simulation to the simulation from TREEKIN whose results are shown in Figure 6. Focussing on the two dominant hishapes [38] and [27], the similarity to the kinetics based on strictly negative structures (Figure 5) is higher than the similarity to the kinetics for the unrestricted approach (Figure 4). By design, the latter retains more detail, which is reflected by the presence of the two not strictly negative hishapes [13,38] and [10.5,38] in this simulation. Again, however, the simulated kinetics is significantly similar to the TREEKIN results. Overall, this result shows that our approach to the simulation of folding kinetics is accurate enough to capture major features of the folding space, such as the late crossing of hishapes [38] and [27].

The c-di-GMP riboswitch of the tfoX from Candidatus desulforudis audaxviator

In the second example, we analysed the c-di-GMP riboswitch of the tfoX gene from Candidatus desulforudis

audaxviator (CP000860.1/c(1860063-1860186), [25]. As shown in Figure 7, it has two states that differ by approximately 2.3 kcal/mol in free energy. The c-di-GMP riboswitches, like all riboswitches, are composed of two domains: an aptamer and an expression platform. The aptamer is more conserved and is responsible for binding c-di-GMP, while the expression platform controls expression by alternative conformations. Here, helix 116.5, which is present in the second hishrep constitutes a Rho-independent terminator hairpin.

We simulated the folding kinetics based on strictly negative hishapes and chose the stable helix ([25.5]) of the aptamer as the initial population (see Figure 8). The hishape [25.5,94.5], which corresponds to the native ON conformation, dominates from $t = 0.5 \mu s$ until thermodynamic equilibrium. Other hishapes such as [7.5,25.5,63.5,94.5,116.5], [25.5,63.5,87,116.5], [25.5,63.5,94.5] and [63.5] appear transiently in different periods. The first two correspond to OFF conformations (helix 116.5 is present), and their fraction is significantly increased from approximately $t = 0.01 \mu s$ to $t = 5,000$



μs . The *hishape* [25.5,63.5,94.5] likely represents a folding intermediate between the ON and OFF conformations, as it is composed of helices from both structures. Its share increases briefly at time point 10,000 μs and drops shortly after, while the fraction of *hishape* [25.5,94.5] increases, which supports the assumption that *hishape* [25.5,63.5,94.5] is a folding intermediate between the ON and OFF conformations. The *hishape* [63.5] appears late ($1e + 06 \mu s$) in the simulation. The short time span ($t = 0.01 \mu s$ to $t = 5,000 \mu s$) where OFF conformations achieve a significant fraction of the folding space reflects the kinetic control of this riboswitch [27]. The folding kinetics restricts the time period during which the RNA is accessible for regulation.

Conclusions

In this paper, we present several methods for improving folding space analysis. First, we introduce strictly negative *hishapes* that represent a reasonable subset of the folding space, i.e., those *hishapes* composed of helices that all have

negative energies. We analysed *hishapes* and their strictly negative variant for correspondence to local optima, and found a large overlap. This result supports our idea of using *hishapes* for folding space analysis. Second, we present HiPATH2, an improved algorithm for calculating suboptimal folding pathways between two given secondary structures. A benchmark confirms that HiPATH2 outperforms its predecessor and other heuristics on the chosen dataset. Finally, we present a new approach for simulating RNA kinetics, which is based on *hishapes* and uses HiPATH2 to compute transition rates. The simulated folding kinetics of two well-studied RNAs show that using our approach allows us to draw functional conclusions. The results for the c-di-GMP riboswitch make us wonder if kinetics can help in identifying new riboswitches. To the best of our knowledge, the existing methods for the identification of riboswitches [19,28-31], are based on sequence and/or secondary structure conservation or on structure comparison. No methods use folding kinetics.

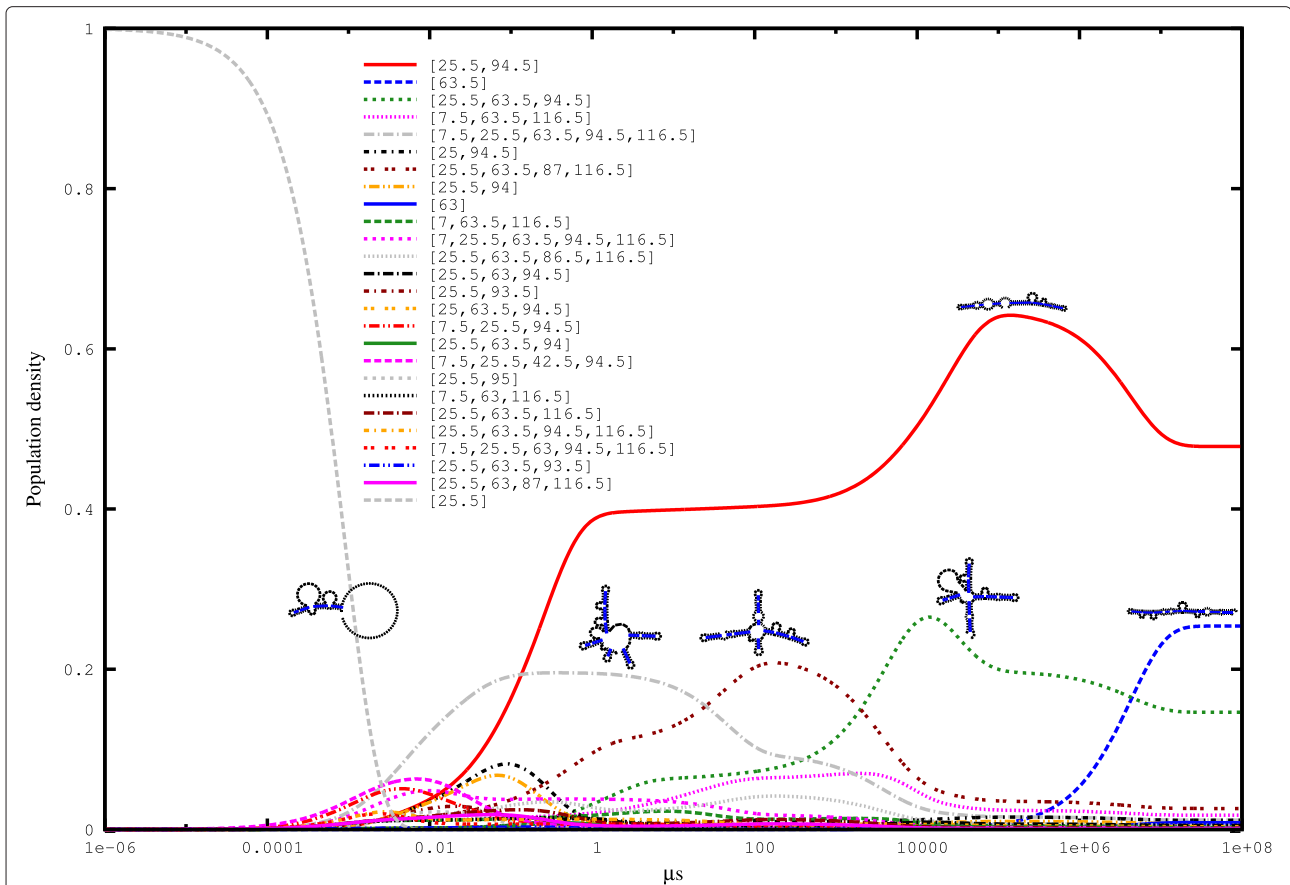


Figure 8 Simulated folding kinetics of the c-di-GMP riboswitch of the *tfoX* gene from *C. desulforudis audaxviator* MP104C. The calculation is based on the 100 best π_n^{SN} hishapes, and we used the stable helix ([25.5]) of the aptamer as the initial population. We show the results for the 25 best hishapes plus hishape [25.5]. The simulation took 24 hours using 64 cores of a 4x AMD Opteron 6282SE machine with 512 GB RAM running under openSuSE 12.2. This figure was generated using: `./HiKinetics.rb -i Input/c_di_GMP_riboswitch.seq -o Output/c_di_GMP_riboswitch_SN -k 100 -t 4 -s -p [25.5]`.

Furthermore, in folding pathways between two conformations, intermediate structures may occur that have helices with slightly different helix indices.

To account for these small variations, we introduce a less stringent version of related hishapes, which we call fuzzy related hishapes.

Definition 1 (Fuzzy related hishapes). Given two hishapes α and β in an arbitrary abstraction type and a user-defined threshold θ , and letting ϕ be a function to extract hairpin loop helix indices, fuzzy related hishapes γ are the hishapes that satisfy

$$\max_{t \in \phi(\gamma)} \min_{z \in \phi(\alpha) \cup \phi(\beta)} |t - z| \leq \theta \quad (1)$$

Restricting the number of fuzzy related hishapes within HiPATH2

The number of (fuzzy related) hishapes has a large impact on the runtime of HiPATH2. For this reason we provide a means to restrict this number. In the previous version

(HiPATH), the calculation of related hishapes always starts at the most abstract level. If, in this level, the number of hishapes is not greater than a user-defined threshold n , the next lower abstraction level is used. This step is performed either until the number of hishapes is greater than n or the user-defined lowest abstraction level t is reached. The number of related hishapes calculated in this way causes a repeated hishape calculation of different abstraction types. For example, if the first attempt does not result in a sufficient number of hishapes, they must be calculated for the next abstraction type, and the initial result will be discarded.

To avoid this issue and speed up HiPATH2, we use an auto-adjust strategy that applies the empirically derived formula shown in Equation 2. Precise asymptotics for the number of abstract shapes have been derived in [15,36] and are defined by $a \times b^n \times n^{-3/2}$ where n is the sequence length. We use this formula to adjust the number of related hishapes for the HiPATH2 calculations. After empirical testing, we chose $a \times b^n = 124,000$. Therefore,

for $n = 500$, k is approximately 10, which means that we keep the 10 fuzzy related *hishapes* with the lowest free energy. This precaution keeps the HiPATH2 calculation within one hour for two *hishapes* of a 500 nt long sequence.

Definition 2 (Auto-adjust fuzzy related *hishape* number).

$$k = 124,000 \times n^{-3/2} \quad (2)$$

HiPATH2 algorithm

For the computation of a single pathway between a given start and target structure, we restrict the search space to fuzzy related *hishapes* as defined by Equation 1. Additionally, given an RNA sequence x , a start structure S and a target structure T , only the shortest path from the start to the target structure is computed. Algorithm 1 shows an outline of HiPATH in pseudocode. In line 4, the N lowest-energy fuzzy related *hishreps* in the π_h abstraction (-t 1) with respect to the helix index list H_U are calculated using RNAHELICES. In line 7, we use a breadth first search (BFS) to estimate the energy barrier between $L[i]$ and $L[j]$, which is stored in the matrix M_{BFS} at position (i, j) . In line 10, we apply a modified version of Dijkstra's algorithm [37] in which the edges are weighted with the barrier energies calculated by the BFS algorithm. Instead of computing the sum of the weights, we take the maximum weight along the path and look for the path with the lowest maximum weight.

Algorithm 1 HiPath2 (rna s , structure S , structure T)

```

1:  $H_S \leftarrow Hishape_x(S), H_T \leftarrow Hishape_x(T) \triangleright Hishape_x$ 
   returns helix indices
2:  $H_U \leftarrow H_S \cup H_T$ 
3:  $N \leftarrow 124,000 * Length(s)^{-3/2}$ 
4:  $L \leftarrow RNAHELICES -R H_U -k N -t 1 \triangleright$  Compute  $N$ 
   fuzzy related (to  $H_U$ ) hishapes
5: for  $i = 1 \rightarrow N$  do
6:   for  $j = 1 \rightarrow N$  do
7:      $M_{BFS}(i, j) \leftarrow BFS_e(L[i], L[j]) \triangleright$  Breadth first
       search returns a barrier energy
8:   end for
9: end for
10:  $maxG \leftarrow LowestPath(L, M_{BFS}) \triangleright$  Dijkstra's algorithm
    (single source, single target)
11: return  $maxG$ 

```

Kinetic folding simulation

In the following section, we describe how the folding kinetics of an RNA molecule is calculated from the barrier energies between *hishapes*. In [4], the authors introduced

a partitioning of the conformation space based on gradient basins of the local energy minima. The authors term these partitions *macrostates* and use the macrostate ensemble free energy to compute the transition rates. In our simulation, we divide the conformation space into *hishapes* \mathcal{H} . For each *hishape* $\alpha \in \mathcal{H}$, we compute the ensemble free energy based on the partition function [38], where $\Delta G(x)$ represents the free energy of conformation x , k is the universal gas constant, and T is the absolute temperature in Kelvin.

$$Z_\alpha = \sum_{x \in \alpha} e^{-\Delta G(x)/kT} \quad (3)$$

and the corresponding *hishape* ensemble energy

$$\Delta G_\alpha = -kT \ln Z_\alpha \quad (4)$$

Between the two *hishapes* α and β , we approximate the transition rates using Arrhenius' equation,

$$r_{\beta\alpha} = A e^{-(\Delta G[\alpha, \beta] - \Delta G(\alpha))/kT} \quad (5)$$

where $\Delta G[\alpha, \beta]$ is the barrier energy between the two *hishreps* of α and β computed by HiPATH2. The pre-exponential factor A can be determined by fitting the available experimental data to the formula $\log k_F = A e^{-aN^b}$, where k_F is the experimentally determined folding rate, and N is the number of nucleotides. In [39], a value of $A = 1.0 \mu s^{-1}$ was proposed, which we use for all our simulations.

Let $p_\alpha(t)$ be the probability of a conformation to be in *hishape* α at time t , and the probability distribution can be computed by the master equation

$$\frac{dp_\alpha(t)}{dt} = \sum_{\beta \in \mathcal{H}} p_\beta(t) r_{\alpha\beta}, \text{ with } r_{\alpha\alpha} = - \sum_{\beta \neq \alpha} r_{\beta\alpha} \quad (6)$$

The equation can be rewritten in matrix form

$$\frac{d}{dt} \vec{p}(t) = R \vec{p}(t) \quad (7)$$

From the matrix differential equation, the folding kinetics are described by (8) where $\vec{p}(0)$ is the initial distribution of the CTMC.

$$\vec{p}(t) = \vec{p}(0) e^{tR} \quad (8)$$

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

JH implemented the software, performed all the computations and drafted the manuscript. BV designed the study and wrote the manuscript. Both authors have read and approved the final manuscript.

Acknowledgements

This work was supported by the German Research Foundation (DFG) (grant Vo 1450/2-1 to BV). The article processing charge was funded by the German Research Foundation (DFG) and the Albert Ludwigs University Freiburg in the funding programme Open Access Publishing.

Received: 9 September 2013 Accepted: 24 February 2014

Published: 28 February 2014

References

1. Flamm C, Fontana W, Hofacker IL, Schuster P: **RNA folding at elementary step resolution.** *RNA* 2000, **6**(3):325–338.
2. Schmitz M, Steger G: **Description of RNA folding by “simulated annealing”.** *J Mol Biol* 1996, **255**(1):254–266.
3. Danilova LV, Pervouchine DD, Favorov AV, Mironov AA: **RNAkinetics: a web server that models secondary structure kinetics of an elongating RNA.** *J Bioinformatics and Comput Biol* 2006, **4**(02):589–596.
4. Wolfinger MT, Svrcek-Seiler WA, Flamm C, Hofacker IL, Stadler PF: **Efficient computation of RNA folding dynamics.** *J Phys A: Math and General* 2004, **37**(17):4731.
5. Cao S, Chen SJ: **Biphasic folding kinetics of RNA pseudoknots and telomerase RNA activity.** *J Mol Biol* 2007, **367**(3):909–924.
6. Tang XY, Thomas S, Tapia L, Giedroc DP, Amato NM: **Simulating RNA folding kinetics on approximated energy landscapes.** *J Mol Biol* 2008, **381**(4):1055–1067.
7. Tang XY, Kirkpatrick B, Thomas S, Song G, Amato NM: **Using motion planning to study RNA folding kinetics.** *J Comput Biol* 2005, **12**(6):862–881.
8. Flamm C, Hofacker IL, Stadler PF, Wolfinger MT: **Barrier trees of degenerate landscapes.** *Z Phys Chem* 2002, **216**(2/2002):155.
9. Mañuch J, Thachuk C, Stacho L, Condon A: **NP-completeness of the energy barrier problem without pseudoknots and temporary arcs.** *Natural Comput* 2011, **10**(1):391–405.
10. Morgan SR, Higgs PG: **Barrier heights between ground states in a model of RNA secondary structure.** *J Phys A-Math Gen* 1998, **31**:3153.
11. Flamm C, Hofacker IL, Maurer-Stroh S, Stadler PF, Zehl M: **Design of multistable RNA molecules.** *RNA* 2001, **7**(2):254–265.
12. Dotu I, Lorenz WA, Hentenyck PV, Clote P: **Computing folding pathways between RNA secondary structures.** *Nucleic Acids Res* 2010, **38**(5):1711–1722.
13. Li Y, Zhang SJ: **Predicting folding pathways between RNA conformational structures guided by RNA stacks.** *BMC Bioinformatics* 2012, **13**(Suppl 3):5.
14. Huang J, Backofen R, Voß B: **Abstract folding space analysis based on helices.** *RNA* 2012, **18**(12):2135–2147.
15. Nebel ME, Scheid A: **On quantitative effects of RNA shape abstraction.** *Theor Biosci* 2009, **128**(4):211–225.
16. Lorenz WA, Clote P: **Computing the partition function for kinetically trapped RNA secondary structures.** *PLoS ONE* 2011, **6**(1):16178.
17. Wakeman CA, Winkler WCIII: **CED: Structural features of metabolite-sensing riboswitches.** *Trends in Biochemical Sciences* 2007, **32**(9):415.
18. Mandal M, Boese B, Barrick JE, Winkler WC, Breaker RR: **Riboswitches control fundamental biochemical pathways in *Bacillus subtilis* and other bacteria.** *Cell* 2003, **113**(5):577–586.
19. Voß B, Meyer C, Giegerich R: **Evaluating the predictability of conformational switching in RNA.** *Bioinformatics* 2004, **20**(10):1573–1582.
20. Xia TB, SantaLucia J, Burkard ME, Kierzek R, Schroeder SJ, Jiao XQ, Cox C, Turner DH: **Thermodynamic parameters for an expanded Nearest-Neighbor model for formation of RNA duplexes with Watson-Crick base pairs.** *Biochemistry-US* 1998, **37**(42):14719–14735.
21. Mathews DH, Sabina J, Zuker M, Turner DH: **Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure.** *J Mol Biol* 1999, **288**(5):911–940.
22. Janssen S, Schudoma C, Steger G, Giegerich R: **Lost in folding space? comparing four variants of the thermodynamic model for RNA secondary structure prediction.** *BMC Bioinformatics* 2011, **12**(1):429.
23. Hofacker IL: **Vienna RNA secondary structure server.** *Nucleic Acids Res* 2003, **31**(13):3429–3431.
24. LeCuyer KA, Crothers DM: **The *Leptomonas collosoma* spliced leader RNA can switch between two alternate structural forms.** *Biochemistry-US* 1993, **32**(20):5301–5311.
25. Weinberg Z, Barrick JE, Yao Z, Roth A, Kim JN, Gore J, Wang JX, Lee ER, Block KF, Sudarsan N, Neph S, Tompa M, Ruzzo WL, Breaker RR: **Identification of 22 candidate structured RNAs in bacteria using the CMfinder comparative genomics pipeline.** *Nucleic Acids Res* 2007, **35**(14):4809–4819.
26. Li Y, Zhang S: **Finding stable local optimal RNA secondary structures.** *Bioinformatics* 2011, **27**(21):2994–3001.
27. Wickiser JK, Winkler WC, Breaker RR, Crothers DM: **The speed of RNA transcription and metabolite binding kinetics operate an FMN riboswitch.** *Molecular Cell* 2005, **18**(1):49–60.
28. Griffiths-Jones S, Moxon S, Marshall M, Khanna A, Eddy SR, Bateman A: **Rfam: annotating non-coding RNAs in complete genomes.** *Nucleic Acids Res* 2005, **33**(suppl 1):121–124.
29. Bengert P, Dandekar T: **Riboswitch finder—a tool for identification of riboswitch RNAs.** *Nucleic Acids Res* 2004, **32**(suppl 2):154–159.
30. Abreu-Goodger C, Merino E: **Ribex: a web server for locating riboswitches and other conserved bacterial regulatory elements.** *Nucleic Acids Res* 2005, **33**(suppl 2):690–692.
31. Chang TH, Huang HD, Wu LC, Yeh CT, Liu BJ, Horng JT: **Computational identification of riboswitches based on RNA conserved functional sequences and conformations.** *RNA* 2009, **15**(7):1426–1430.
32. Mathews DH, Disney MD, Childs JL, Schroeder SJ, Zuker M, Turner DH: **Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure.** *Proc Natl Acad Sci USA* 2004, **101**(19):7287–7292.
33. Sauthoff G, Janssen S, Giegerich R: **Bellman’s gap: a declarative language for dynamic programming.** In *Proceedings of the 13th International ACM SIGPLAN Symposium on Principles and Practices of Declarative Programming*. New York, NY, USA: ACM; 2011:29–40.
34. Giegerich R, Sauthoff G: **Yield grammar analysis in the Bellman’s GAP compiler.** In *Proceedings of the Eleventh Workshop on Language Descriptions, Tools and Applications*. New York, NY, USA: ACM; 2011:7.
35. Sauthoff G, Möhl M, Janssen S, Giegerich R: **Bellman’s GAP—a language and compiler for dynamic programming in sequence analysis.** *Bioinformatics* 2013, **29**(5):551–560.
36. Lorenz WA, Ponty Y, Clote P: **Asymptotics of RNA shapes.** *J Comput Biol* 2008, **15**(1):31–63.
37. Dijkstra EW: **A note on two problems in connexion with graphs.** *Numer Math* 1959, **1**(1):269–271.
38. McCaskill JS: **The equilibrium partition function and base pair binding probabilities for RNA secondary structure.** *Biopolymers* 1990, **29**(6-7):1105–1119.
39. Hyeon CB, Thirumalai D: **Chain length determines the folding rates of RNA.** *Biophysical J* 2012, **102**(3):11–13.

doi:10.1186/1471-2105-15-60

Cite this article as: Huang and Voß: Analysing RNA-kinetics based on folding space abstraction. *BMC Bioinformatics* 2014 **15**:60.

Submit your next manuscript to BioMed Central and take full advantage of:

- Convenient online submission
- Thorough peer review
- No space constraints or color figure charges
- Immediate publication on acceptance
- Inclusion in PubMed, CAS, Scopus and Google Scholar
- Research which is freely available for redistribution

Submit your manuscript at
www.biomedcentral.com/submit

